Delft University of Technology

Fluid Structure Interaction Assignment 2

Delft University of Technology, Delft, South Holland, 2628 CD

Changkyu Park 4646061
Malte Wegener 4672194

Submission Date: March 17, 2021
Course: AE4117

Contents

1. Geometric Conservation Law (GCL)

1.1. Exercise 1.1

Starting with the GCL

$$\frac{d}{dt}\int_{\Omega_i}\mathbf{U}dV + \oint_{\partial\Omega_i}\left[\mathbf{F}(\mathbf{U}) - \mathbf{U}\frac{d\mathbf{x}}{dt}\right]\cdot\mathbf{n}dS = 0, \tag{1.1}$$

$\mathbf{U} = \mathbf{U}_0$ is assumed to be constant and $\mathbf{F}(\mathbf{U}_0)$ is assumed to be constant as well. Substituting into (1.1) yields

$$\frac{d}{dt}\int_{\Omega_i}\mathbf{U}_0 dV + \oint_{\partial\Omega_i}\left[\mathbf{F}(\mathbf{U}_0) - \mathbf{U}_0\frac{d\mathbf{x}}{dt}\right]\cdot\mathbf{n}dS = 0. \tag{1.2}$$

As $\int_S C\cdot\mathbf{n}dS = 0$ for every closed surface S, (1.2) simplifies to

$$\mathbf{U}_0\frac{d}{dt}\int_{\Omega_i}dV - \mathbf{U}_0\oint_{\partial\Omega_i}\frac{d\mathbf{x}}{dt}\cdot\mathbf{n}dS = 0. \tag{1.3}$$

Assuming $\mathbf{U}_0 \neq \mathbf{0}$, (1.3) becomes

$$\frac{d}{dt}\int_{\Omega_i}dV = \oint_{\partial\Omega_i}\frac{d\mathbf{x}}{dt}\cdot\mathbf{n}dS. \tag{1.4}$$

1.2. Exercise 1.2

The semi-discrete geometric conservation law given as

$$\frac{d}{dt}V_i = \sum_{\text{faces}}\left(\frac{d\mathbf{x}}{dt}\cdot\mathbf{n}S\right)_{\text{face}}. \tag{1.5}$$

The right-hand side corresponds to a finite volume discretization in space. Each component of the sum represents the volume/surface swept by a boundary face of the finite volume. The left-hand side of the equation represents the change in volume of the considered finite volume.

1.3. Exercise 1.3

Starting from the semi-discrete geometric conservation law given as

$$\frac{d}{dt}V_i = \sum_{j=1}^{\text{faces}}\left(\frac{d\mathbf{x}}{dt}\cdot\mathbf{n}S\right)_j \tag{1.6}$$

Considering the BDF2 time integration scheme expressed as

$$\frac{\alpha_1\phi^{n+1} + \alpha_2\phi^n + \alpha_3\phi^{n-1}}{\Delta t} + R\left(\phi^{n+1}\right) = 0, \tag{1.7}$$

the fully discrete geometric conservation law for the volume i is formed as

$$\frac{\alpha_1 V_i^{n+1} + \alpha_2 V_i^n + \alpha_3 V_i^{n-1}}{\Delta t} = \sum_{j=1}^{\text{faces}}\left(\frac{d\mathbf{x}}{dt}\cdot\mathbf{n}\Delta S\right)_{i,j}^{n+1}. \tag{1.8}$$

For BDF2, the coefficients α are given as $\left[\frac{3}{2}, -2, \frac{1}{2}\right]$, thus the D-GCL can be simplified to

$$\frac{3}{2}\left(\frac{V_i^{n+1} - V_i^n}{\Delta t}\right) - \frac{1}{2}\left(\frac{V_i^n - V_i^{n-1}}{\Delta t}\right) = \sum_{j=1}^{\text{faces}}\left(\frac{d\mathbf{x}}{dt}\cdot\mathbf{n}\Delta S\right)_{i,j}^{n+1}. \tag{1.9}$$

It is assumed that the D-GCL is not only satisfied for the volume but for every face, which reduces it to

$$\frac{3}{2}\frac{\Delta V_{i,j}^{n+1}}{\Delta t} - \frac{1}{2}\frac{V_{i,j}^n}{\Delta t} = \left(\frac{d\mathbf{x}}{dt}\cdot\mathbf{n}\Delta S\right)_{i,j}^{n+1}. \tag{1.10}$$

1.4. Exercise 1.4

Starting from the semi-discrete geometric conservation law is given as

$$\frac{d}{dt}V_i = \sum_{j=1}^{\text{faces}} \left(\frac{d\mathbf{x}}{dt} \cdot \mathbf{n}S \right)_j , \tag{1.11}$$

the ESDIRK integration method is considerd and it is expressed as

$$\frac{\phi^{(k)} - \phi^n}{\Delta t} + \sum_{j=1}^{k} a_{kj} R\left(\phi^{(j)}\right) = 0. \tag{1.12}$$

The D-GCL can then be written as

$$\frac{V_i^k - V_i^n}{\Delta t} = \sum_{j=1}^{k} a_{kj} \sum_{l=1}^{\text{faces}} \left(\frac{d\mathbf{x}}{dt} \cdot \mathbf{n}S \right)_{i,l}^{j} . \tag{1.13}$$

Assuming that the D-GCL is satisfied for every face of the volume and separating known from unknown terms, the D-GCL can be written as

$$\frac{V_{i,l}^k - V_{i,l}^n}{\Delta t} - \sum_{j=1}^{k-1} a_{kj} \left(\frac{d\mathbf{x}}{dt} \cdot \mathbf{n}S \right)_{i,l}^{j} = a_{kk} \left(\frac{d\mathbf{x}}{dt} \cdot \mathbf{n}S \right)_{i,l}^{k} . \tag{1.14}$$

As for ESDIRK $a_{11} = 0$ and thus the first stage is the same as the final result of the previous step, $\left(\frac{d\mathbf{x}}{dt} \cdot \mathbf{n}S \right)_{i,l}^{k}$ can be any real number and thus $\left(\frac{d\mathbf{x}}{dt} \right)_{i,l}^{k}$ can be any vector. For all later stages, the D-GCL for a face l of a volume i is given by

$$\frac{1}{a_{kk}} \left[\frac{V_{i,l}^k - V_{i,l}^n}{\Delta t} - \sum_{j=1}^{k-1} a_{kj} \left(\frac{d\mathbf{x}}{dt} \cdot \mathbf{n}S \right)_{i,l}^{j} \right] = \left(\frac{d\mathbf{x}}{dt} \cdot \mathbf{n}S \right)_{i,l}^{k} . \tag{1.15}$$

2. Simulation of uniform flow on a one-dimensional moving mesh

2.1. Exercise 2.1

To analyse the error of methods 1 and 2, a plot has been made for each method with the 3 different values of Δt which are 0.1, 0.05 and 0.025 using the *dgcl_BE.m* program. They are shown in Figure 1 and Figure 2 for method 1 and 2 respectively.

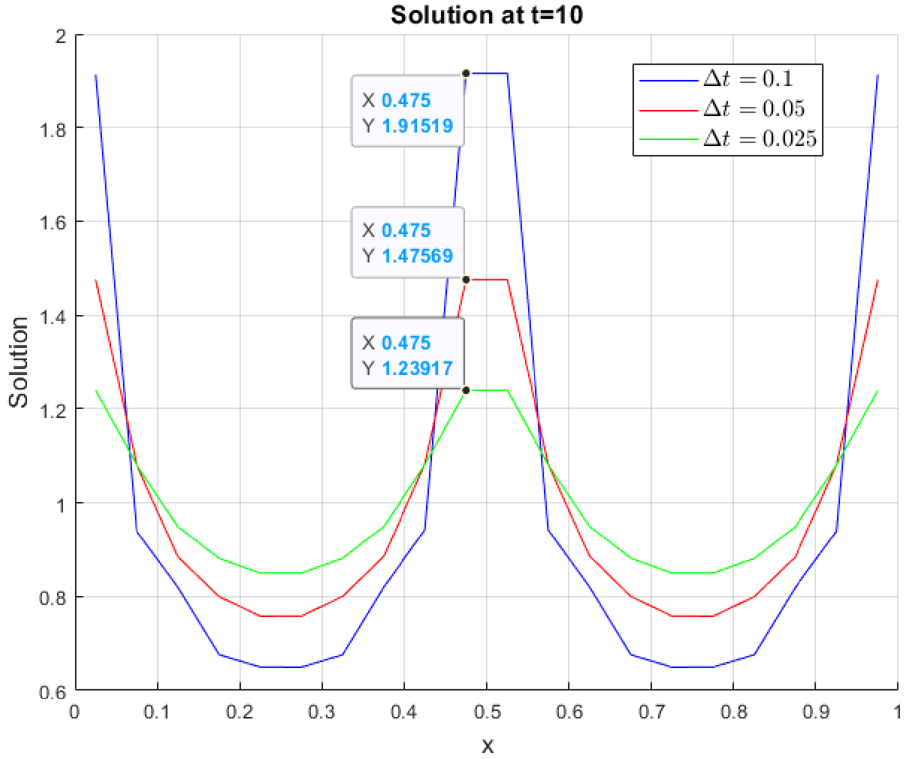

Fig. 1 Results for Method 1 for Δ = 0.1, 0.05 and 0.025

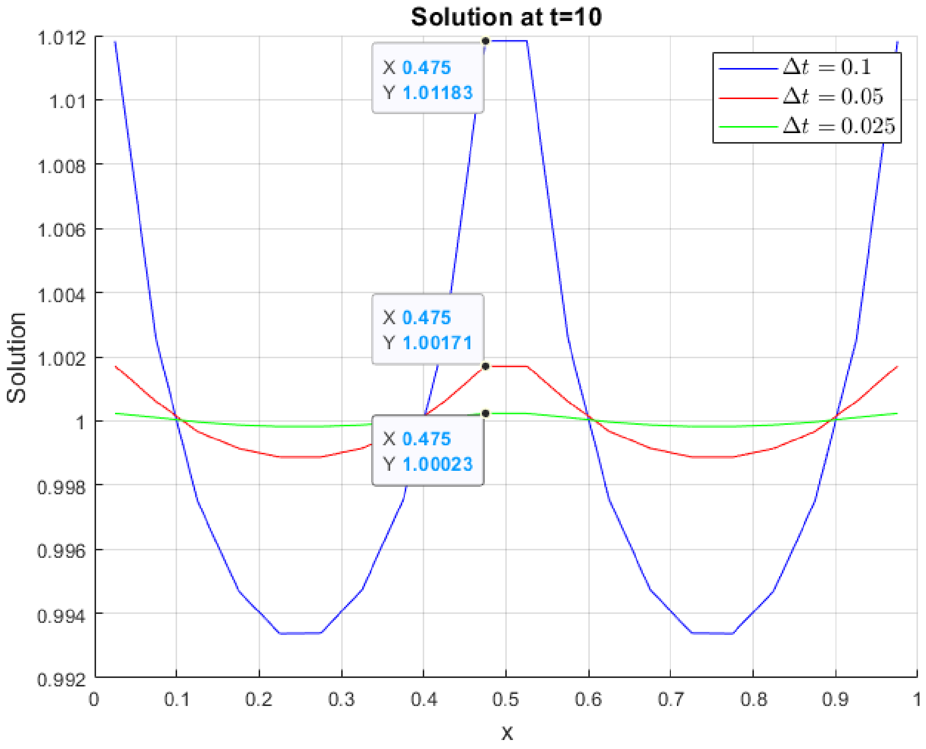

Fig. 2 Results for Method 2 for Δ = 0.1, 0.05 and 0.025

Firstly for method 1 shown in Figure 1, it can be observed that the amplitude decreases with decreasing time step Δt. Defining the error ϵ as the difference between this amplitude and the exact solution which is unity, the order of accuracy can be found. The errors for the 3 time steps are

$$\begin{aligned} \Delta t = 0.1,\ \ \epsilon &= 1.9152 - 1 = 0.9152 \\ \Delta t = 0.05,\ \ \epsilon &= 1.4757 - 1 = 0.4757 \\ \Delta t = 0.025,\ \ \epsilon &= 1.2392 - 1 = 0.2392. \end{aligned}$$

The error decreases to approximately 52% of the initial error when the time step Δt is halved going from $\delta t = 0.1$ to $\delta t = 0.05$. For the time step going from $\delta t = 0.05$ to $\delta t = 0.025$, the error decreases to approximately 50%. Thus, the order of accuracy is about 1 for the BE scheme of method 1.

A similar analysis can be done for method 2. The errors for the 3 time steps are

$$\begin{aligned} \Delta t = 0.1,\ \ \epsilon &= 1.01183 - 1 = 0.01183 \\ \Delta t = 0.05,\ \ \epsilon &= 1.00171 - 1 = 0.00171 \\ \Delta t = 0.025,\ \ \epsilon &= 1.00023 - 1 = 0.00023. \end{aligned}$$

The error decreases to approximately 15% of the initial error when the time step Δt is halved going from $\delta t = 0.1$ to $\delta t = 0.05$. For the time step going from $\delta t = 0.05$ to $\delta t = 0.025$, the error decreases to approximately 13%. Thus, the order of accuracy is about 3 for the BE scheme of method 2. In all, we have

$$\begin{aligned} &\text{Method 1: } O(\Delta t) \\ &\text{Method 2: } O(\Delta t^3) \end{aligned}$$

2.2. Exercise 2.2

Now, two types of errors, ϵ_1 and ϵ_2, are to be considered. ϵ_1 is the root-mean-square error at the final time where as ϵ_2 is the root-mean-square error over the entire simulation. They are expressed as

$$\begin{aligned} \epsilon_1 &= \sqrt{\frac{\sum_{i=1}^{N}\left(u_i^M - u_{ex}\left(x_i, t_M\right)\right)^2}{N}} \qquad \text{and} \\ \epsilon_2 &= \sqrt{\frac{\sum_{j=1}^{M}\sum_{i=1}^{N}\left(u_i^j - u_{ex}\left(x_i, t_j\right)\right)^2}{NM}}, \end{aligned}$$

where u_{ex} is the exact solution, N is the total number of spatial unknowns and M is the total number of time steps taken. For this analysis, the time steps Δt are taken to be

$$\Delta t = 0.1 \times 2^{-k},$$

where $k = 0, 1, 2, 3, 4, 5, 6$. The two errors and the time step have been implemented into the *dgcl_BE.m* program and snippets of it containing **only the altered or additional part to the original program** have been shown below.

```
%% Input data
dt_i  = 0.1;    % base time step
kend = 6;
e1   = zeros(1,kend+1);
e2   = zeros(1,kend+1);
%% Simulation loop
for k=0:kend
%% Initialisation
  dt = dt_i*2^(-1*k);
  M  = size(dt:dt:tend,2); % Number of time steps
e2sub  = zeros(1,M);
```

```
counter = 0;
for t=dt:dt:tend
  % First error
  if abs(t-tend)<0.001
     error1 = sum((u - u_tn).^2)/N;
     e1(k+1) = error1;
  end
  % Second error
  sum1          = sum((u - u_tn).^2);
  counter       = counter + 1;
  e2sub(counter) = sum1;
  if abs(t-tend)<0.001
     error2 = sum(e2sub)/N/M;
     e2(k+1) = error2;
  end
end
```

Logarithms of the two errors, ϵ_1 and ϵ_2, and the time step Δt have been taken to visualise the order of accuracy for method 1, 2 and 3. These were then plotted as shown in Figure 3, Figure 4 and Figure 5 for method 1, 2 and 3 respectively.

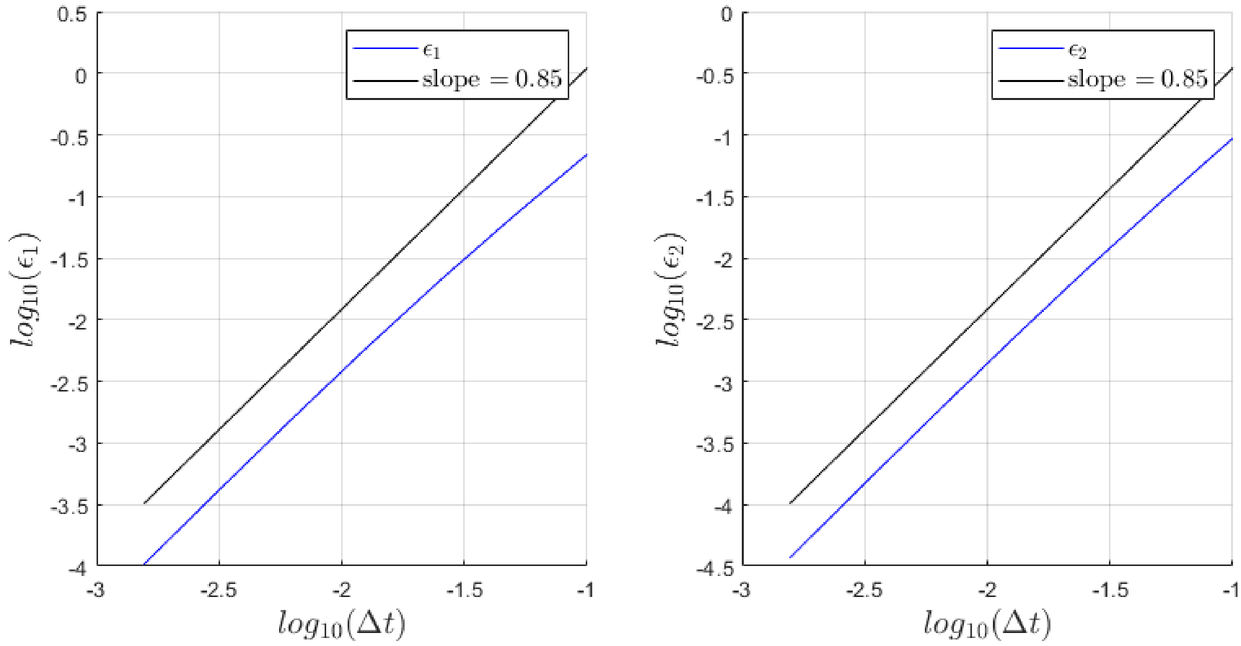

Fig. 3 Errors of method 1

Firstly, for method 1 which is shown in Figure 3, it can be observed that both the logarithm of its ϵ_1 and ϵ_2 follow a slope of 0.85. This translates to an order of accuracy of 0.85, $O(\Delta t^{0.85})$.

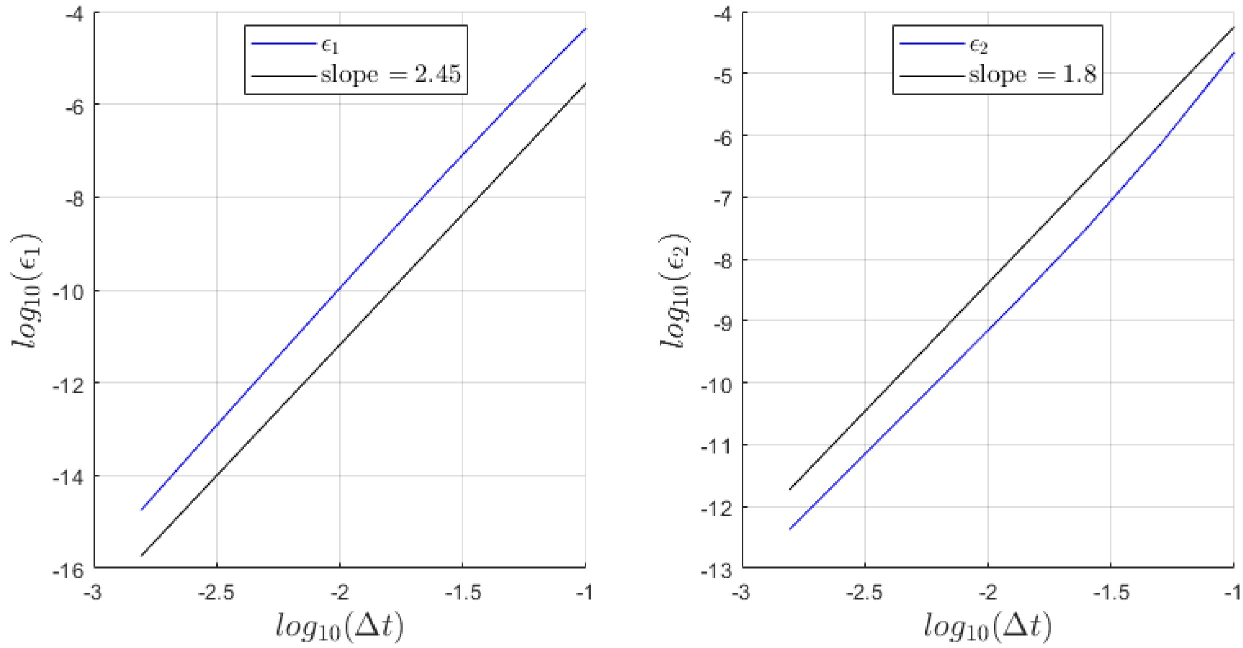

Fig. 4 Errors of method 2

As for method 2 shown in Figure 4, slope of 2.45 and 1.8 for logarithm of ϵ_1 and ϵ_2 can be observed respectively which translates to an order of accuracy of 2.45, $O(\Delta t^{2.45})$ and 1.8, $O(\Delta t^{1.8})$.

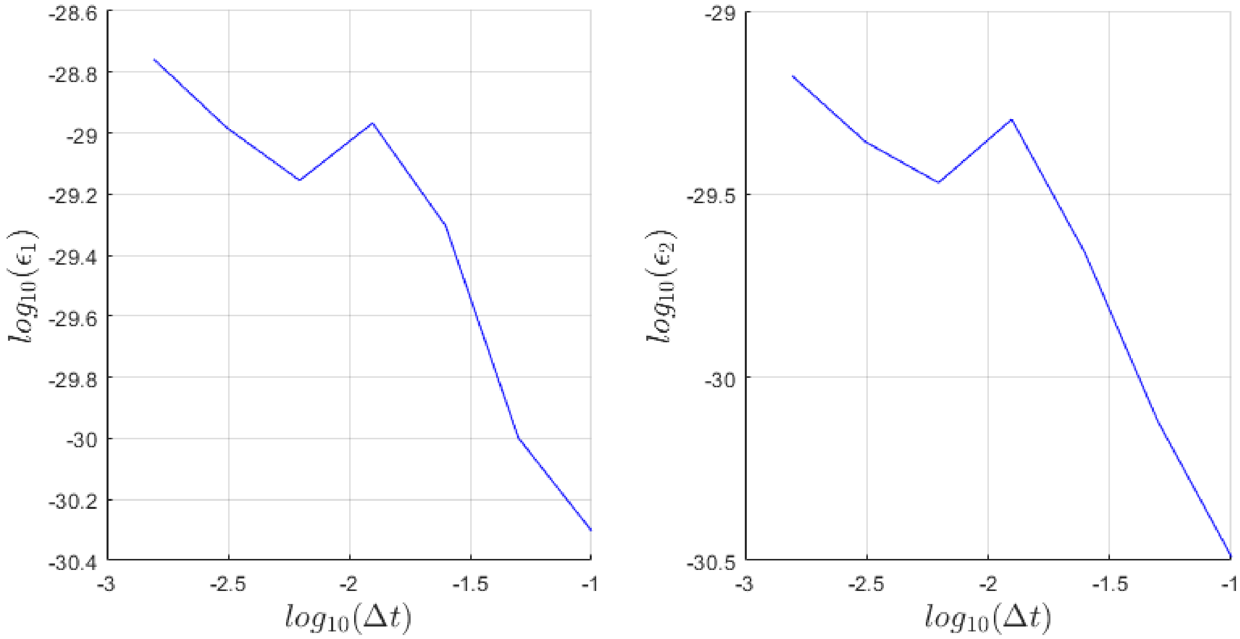

Fig. 5 Errors of method 3

Lastly for method 3 which satisfies the D-GCL, the logarithm of the errors shown in Figure 5 have values around -29 which translates to errors of $\epsilon_1, \epsilon_2 \approx 10^{-29}$ which is a significant amount below the machine epsilon. Thus, they can be treated as absolute zeros. However, it is to be noted that error for this method increases with decreasing Δt which could be due to the rounding errors of the program piling up with the larger number of time steps.

2.3. Exercise 2.3

For this exercise, the error for method 1 and 2 are defined as

$$\begin{aligned}
\epsilon &= \frac{x^{n+1} - x^{n}}{\Delta t} - \left.\frac{dx}{dt}\right|_{t_{n+1}} \quad \text{and} \\
\epsilon &= \frac{x^{n+1} - x^{n}}{\Delta t} - \left.\frac{dx}{dt}\right|_{t_{n+1/2}} \qquad (2.1)
\end{aligned}$$

respectively. The following is the provided approximation of error for method 1:

$$\begin{aligned}
\epsilon &= \left[\left.\frac{dx}{dt}\right|_{t_{n+1}} - \left.\frac{dx}{dt}\right|_{t_{n+1}} \right] - \frac{\Delta t}{2} \left.\frac{d^2 x}{dt^2}\right|_{t_{n+1}} + O\left(\Delta t^2\right) \\
&= O(\Delta t),
\end{aligned}$$

which shows that method 1 is first order accurate.

For method 2, the Taylor series expansion for x^n and x^{n+1} about $t_{n+1/2}$ are required to accommodate the exact mesh velocity at $t_{n+1/2}$. Firstly, the Taylor series expansion for x^n about $t_{n+1/2}$ is

$$x^n = x^{n+1/2} - \frac{\Delta t}{2} \left.\frac{dx}{dt}\right|_{t_{n+1/2}} + \frac{\Delta t^2}{8} \left.\frac{d^2 x}{dt^2}\right|_{t_{n+1/2}} - \frac{\Delta t^3}{48} \left.\frac{d^3 x}{dt^3}\right|_{t_{n+1/2}} + \frac{\Delta t^3}{384} \left.\frac{d^4 x}{dt^4}\right|_{t_{n+1/2}} + O(\Delta t^5). \qquad (2.2)$$

As for the Taylor series expansion for x^{n+1} about $t_{n+1/2}$, it is expressed as

$$x^{n+1} = x^{n+1/2} + \frac{\Delta t}{2} \left.\frac{dx}{dt}\right|_{t_{n+1/2}} + \frac{\Delta t^2}{8} \left.\frac{d^2 x}{dt^2}\right|_{t_{n+1/2}} + \frac{\Delta t^3}{48} \left.\frac{d^3 x}{dt^3}\right|_{t_{n+1/2}} + \frac{\Delta t^3}{384} \left.\frac{d^4 x}{dt^4}\right|_{t_{n+1/2}} + O(\Delta t^5). \qquad (2.3)$$

Substituting (2.2) and (2.3) into (2.1), the approximation of error for method 2 can be found and is expressed as

$$\begin{aligned}
\epsilon &= \left[\left.\frac{dx}{dt}\right|_{t_{n+1}} - \left.\frac{dx}{dt}\right|_{t_{n+1}} \right] - \frac{\Delta t^2}{24} \left.\frac{d^3 x}{dt^3}\right|_{t_{n+1/2}} + O\left(\Delta t^4\right) \\
&= O(\Delta t^2),
\end{aligned}$$

which shows that method 2 is second order accurate.

It is realised that the orders of accuracy derived from numerical results in Exercise 2.2 closely represent the analytical results found above. For method 1, since both errors, ϵ_1 and ϵ_2, showed an order of 0.85, it cannot be determined which is more precise. However, for method 2, ϵ_1 had an order of 2.45 whereas ϵ_2 had an order of 1.8. Since 1.8 is closer to 2 than 2.45, **the second error ϵ_2 is more precise**.

2.4. Exercise 2.4

The second order Backward Differencing scheme (BDF2) is shown in (1.7) which is evaluated at implicit time level t_{n+1}. The implementation into *dgcl_BDF2.m* program is shown in the following.

```
%% IMPLEMENTATION OF DGCL
dxidt_dgcl  = (alpha(1)*xi + alpha(2)*xi_tn + alpha(3)*xi_tnm1) / dt;
```

Just like it was done for Exercise 2.2, the two errors, ϵ_1 and ϵ_2, have been implemented into the program the exact same way. Then, the logarithms of these errors and the time step Δt have been taken to visualise the order of accuracy for method 1, 2 and 3. These were then plotted as shown in Figure 6, Figure 7 and Figure 8 for method 1, 2 and 3 respectively.

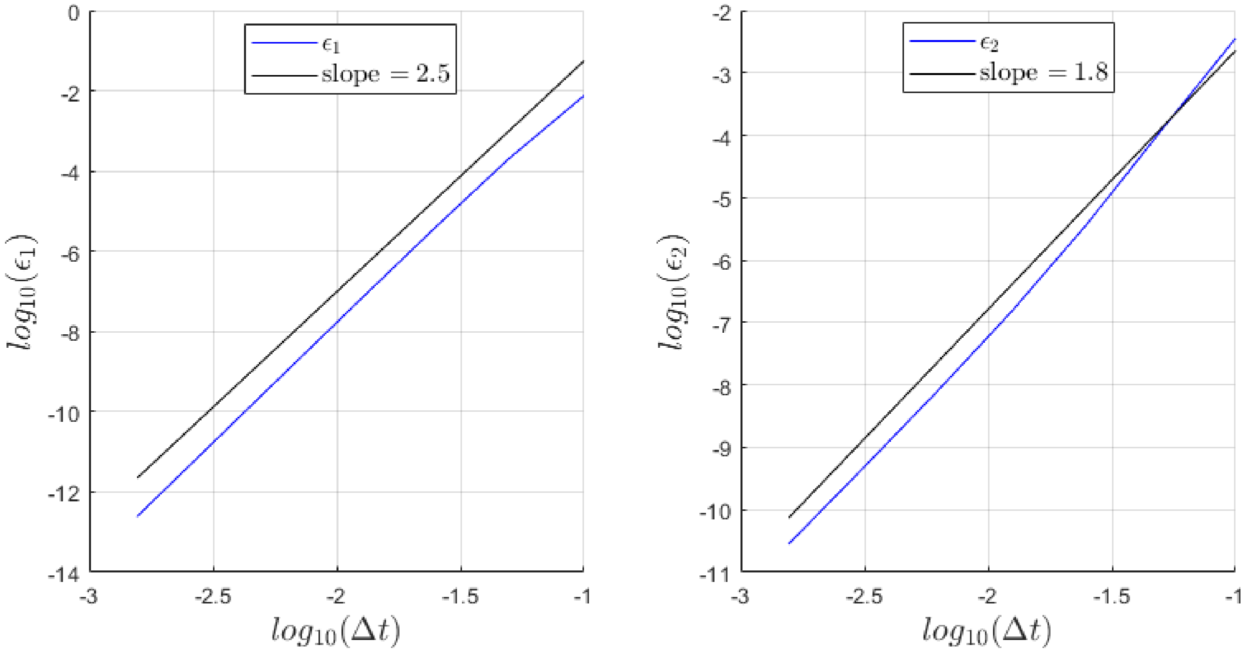

Fig. 6 BDF2: Errors of method 1

Firstly, for method 1 shown in Figure 6, slope of 2.5 and 1.8 for logarithm of ϵ_1 and ϵ_2 can be observed respectively which translates to an order of accuracy of 2.5, $O(\Delta t^{2.5})$ and 1.8, $O(\Delta t^{1.8})$.

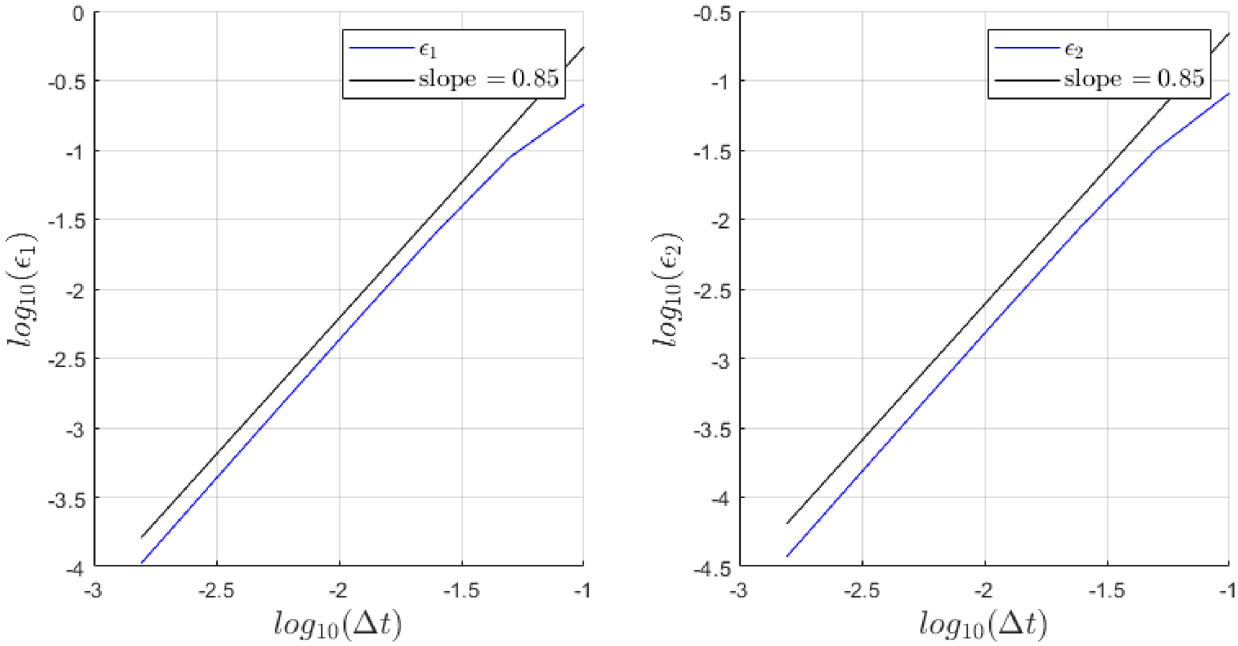

Fig. 7 BDF2: Errors of method 2

As for method 2 which is shown in Figure 7, it can be observed that both the logarithm of its ϵ_1 and ϵ_2 follow a slope of 0.85. This translates to an order of accuracy of 0.85, $O(\Delta t^{0.85})$.

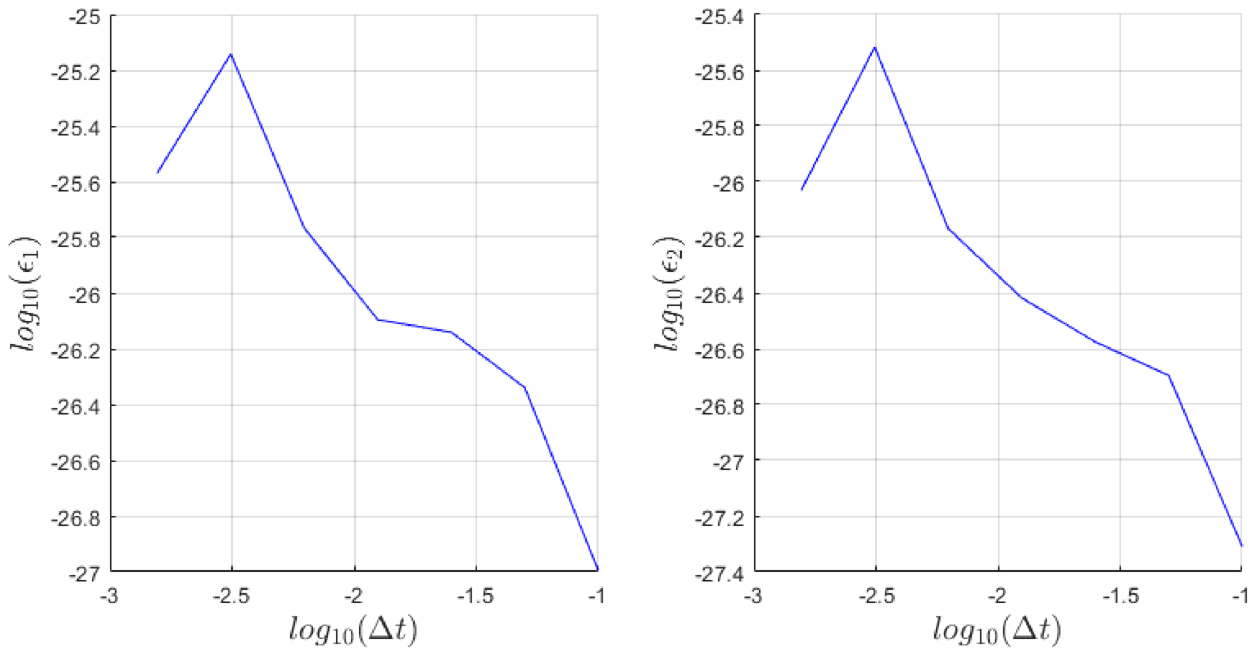

Fig. 8 BDF2: Errors of method 3

Lastly, for method 3 which satisfies the D-GCL, a similar conclusion can be made as for the BE method shown in Figure 5. The logarithm of the errors shown in Figure 8 have values around -26 which translates to errors of $\epsilon_1, \epsilon_2 \approx 10^{-26}$ which is a significant amount below the machine epsilon. Thus, they can be treated as absolute zeros. However, it is to be noted that error for this method increases with decreasing Δt which could be due to the rounding errors of the program piling up.

Like it was done for Exercise 2.3, the error for when using the exact mesh velocity at t_{n+1} BDF2 scheme can be expressed as

$$\epsilon = \frac{3x^{n+1} - 4x^n + x^{n-1}}{2\Delta t} - \left.\frac{dx}{dt}\right|_{t_{n+1}}, \tag{2.4}$$

where as for exact mesh velocity at $t_{n+1/2}$ is used, the error is

$$\epsilon = \frac{3x^{n+1} - 4x^n + x^{n-1}}{2\Delta t} - \left.\frac{dx}{dt}\right|_{t_{n+1/2}}. \tag{2.5}$$

First, the Taylor series expansion of x around t_{n+1} is evaluated for x^{n-1}

$$x^{n-1} = x^{n+1} - 2\Delta t \left.\frac{dx}{dt}\right|_{t_{n+1}} + 2\Delta t^2 \left.\frac{d^2x}{dt^2}\right|_{t_{n+1}} - \frac{4\Delta t^3}{3} \left.\frac{d^3x}{dt^3}\right|_{t_{n+1}} + \frac{2\Delta t^4}{3} \left.\frac{d^4x}{dt^4}\right|_{t_{n+1}} + O(\Delta t^5). \tag{2.6}$$

Secondly, the Taylor series expansion of x around t_{n+1} evaluated for x^n is taken from the assignment document and it is

$$x^n = x^{n+1} - \Delta t \left.\frac{dx}{dt}\right|_{t_{n+1}} + \frac{\Delta t^2}{2} \left.\frac{d^2x}{dt^2}\right|_{t_{n+1}} - \frac{\Delta t^3}{6} \left.\frac{d^3x}{dt^3}\right|_{t_{n+1}} + \frac{\Delta t^4}{24} \left.\frac{d^4x}{dt^4}\right|_{t_{n+1}} + O\left(\Delta t^5\right). \tag{2.7}$$

Substituting (2.6) and (2.7) into (2.4), we get

$$\begin{aligned} \epsilon &= \left[\left.\frac{dx}{dt}\right|_{t_{n+1}} - \left.\frac{dx}{dt}\right|_{t_{n+1}} \right] - \frac{\Delta t^2}{3} \left.\frac{d^3x}{dt^3}\right|_{t_{n+1}} + O(\Delta t^3) \\ &= O(\Delta t^2). \end{aligned}$$

Thus, it can be said that for the BDF2 scheme, method 1 has an order of accuracy of 2. From Figure 6, it can be seen that the second error ϵ_2 is much closer to this analytical result with a slope of 1.8 compared to the first error ϵ_1 that has a slope of 2.45 which is a lot further away to 2 than 1.8.

Next, for (2.5), the Taylor series expansion of x^{n-1} around $t_{n+1/2}$ is required. It is expressed as

$$x^{n-1} = x^{n+1/2} - \frac{3\Delta t}{2}\left.\frac{dx}{dt}\right|_{t_{n+1/2}} + \frac{9\Delta t^2}{8}\left.\frac{d^2x}{dt^2}\right|_{t_{n+1/2}} - \frac{9\Delta t^3}{16}\left.\frac{d^3x}{dt^3}\right|_{t_{n+1/2}} + \frac{81\Delta t^4}{384}\left.\frac{d^4x}{dt^4}\right|_{t_{n+1/2}} + O(\Delta t^5) \tag{2.8}$$

Substituting (2.2), (2.3) and (2.8) into (2.5), we have

$$\begin{aligned}\epsilon &= \left[\left.\frac{dx}{dt}\right|_{t_{n+1/2}} - \left.\frac{dx}{dt}\right|_{t_{n+1/2}}\right] - \frac{\Delta t}{2}\left.\frac{d^2x}{dt^2}\right|_{t_{n+1/2}} + O(\Delta t^2)\\ &= O(\Delta t).\end{aligned}$$

Thus, analytically, method 2 for BDF2 scheme has an order of accuracy of 1. From Figure 7, both the first and second error have slopes of 0.85. Hence, with the second error better-representing method 1, it can be concluded that **the second error ϵ_2 is more precise**.

3. Simulation of uniform flow on a moving two-dimensional mesh

3.1. Exercise 3.1

The volume swept by a face ΔV^{n+1}_{face} is given as

$$\Delta V^{n+1}_{face} = (\boldsymbol{c}^{n+1} - \boldsymbol{c}^{n}) \cdot \frac{\boldsymbol{n}S^{n+1} + \boldsymbol{n}S^{n}}{2} \tag{3.1}$$

according to [1]. For a backward Euler time integration scheme, the computed mesh velocity has to satisfy the D-GCL given as

$$\left(\frac{d\mathbf{x}}{\mathrm{d}t} \cdot \mathbf{n}\right)^{n+1} = \frac{\Delta V^{n+1}_{\text{face}}}{\Delta t S^{n+1}}. \tag{3.2}$$

The right hand side of the equation is known and the unit vector $\mathbf{n}^{n+1}$ is of known length, thus it is known that the velocity vector $\frac{d\mathbf{x}}{dt}^{n+1}$ can be expressed as

$$\frac{d\mathbf{x}}{dt}^{n+1} = \frac{\Delta V^{n+1}_{\text{face}}}{\Delta t S^{n+1}}\mathbf{n} + C\mathbf{v} \tag{3.3}$$

where C is any constant and $\mathbf{v} \in \left\{\mathbb{R}^2 | \mathbf{v} \cdot \mathbf{n} = 0\right\}$. As all vectors of this form satisfy the D-GCL, a single vector has to be chosen, the trivial choice is $C = 0$, which is used in the assignment. The chosen velocity computation is implemented in the MATLAB code as

```
%% IMPLEMENT THE D-GCL HERE %%
dv = (c_TNP1(IDface,:) - c_TN(IDface,:)) * (Sn_TNP1(IDface,:)+Sn_TN(IDface,:))'/dt/2;
velocity = Sn_TNP1(IDface,:) * dv/((Sn_TNP1(IDface,2)^2+Sn_TNP1(IDface,1)^2));
meshVelocity(IDcell,IDface,1) = velocity(1);
meshVelocity(IDcell,IDface,2) = velocity(2);
```

The computation is run for 2 periods with different numbers of timesteps per period and the error defined as the absolute deviation from a uniform solution is recorded and plotted in Figure 9.

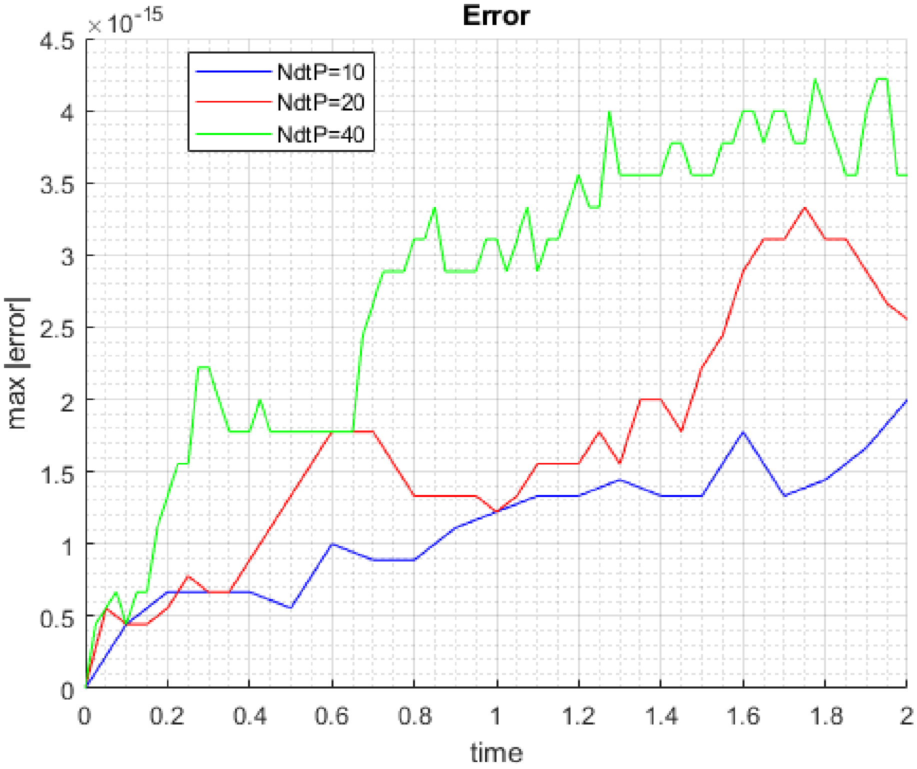

Fig. 9 Comparison of errors for D-GCL for different number of timesteps per period (NdtP)

It can be seen that the error at every time stage is of $O(10^{-15})$ and growing slowly. This error is close to machine epsilon around 1, thus the solution is uniformly 1 over the domain. The small round-off error occurring every iteration however accumulates and thus the error increases with the number of iterations.

3.2. Exercise 3.2

The volume swept by a face ΔV_{face}^k for a stage in a multi-stage method is given by a similar expression as in Exercise 3.1 as

$$\Delta V_{face}^k = (\boldsymbol{c}^k - \boldsymbol{c}^n) \cdot \frac{\boldsymbol{n}S^k + \boldsymbol{n}S^n}{2}. \tag{3.4}$$

As shown in exercise 1.2, the velocity for the first stage of the ESDIRK integration technique can be chosen as the 0 vector. For all other stages, the following D-GCL has to be satisfied which is expressed as

$$\left(\frac{d\mathbf{x}}{\mathrm{d}t} \cdot \mathbf{n}\right)^k = \frac{1}{a_{kk}S^k}\left[\frac{\Delta V_{\text{face}}^k}{\Delta t} - \sum_{j=1}^{k-1} a_{kj}\left(\frac{d\mathbf{x}}{\mathrm{d}t} \cdot \mathbf{n}S\right)^j\right] \tag{3.5}$$

where $\left(\frac{d\mathbf{x}}{\mathrm{d}t} \cdot \mathbf{n}S\right)^j$ is known from previous stages. Similarly to the previous exercise, a vector satisfying the D-GCL is given by

$$\frac{d\mathbf{x}}{dt}^k = \frac{1}{a_{kk}S^k}\left[\frac{\Delta V_{\text{face}}^k}{\Delta t} - \sum_{j=1}^{k-1} a_{kj}\left(\frac{d\mathbf{x}}{\mathrm{d}t} \cdot \mathbf{n}S\right)^j\right]\mathbf{n} + C\mathbf{v} \tag{3.6}$$

where C is any constant and $\mathbf{v} \in \left\{\mathbb{R}^2 | \mathbf{v} \cdot \mathbf{n} = 0\right\}$. For the implementation, C is chosen to be 0. The computation of the mesh velocity is implemented as

```
if (k==1)
    meshVelocity(IDcell,IDface,1) = 0;%velocity(1);
    meshVelocity(IDcell,IDface,2) = 0;%velocity(2);
else
    sm = 0;
    for j=1:k-1
        a = Butcher(k,j) * snk(IDcell,IDface,j);
        sm = sm + a;
    end
    akk = Butcher(k,k);
    dv = (c_TNP1(IDface,:) - c_TN(IDface,:)) * (Sn_TNP1(IDface,:)+Sn_TN(IDface,:))'/dt/2;
    mag = sqrt(Sn_TNP1(IDface,2)^2+Sn_TNP1(IDface,1)^2);
    velocity = 1/akk*Sn_TNP1(IDface,:)/(mag^2)*(dv-sm);
    meshVelocity(IDcell,IDface,1) = velocity(1);
    meshVelocity(IDcell,IDface,2) = velocity(2);
end
```

The implementation is then tested for different number of time steps per period over two periods, while the error is recorded for every time step. The results can be seen in Figure 10.

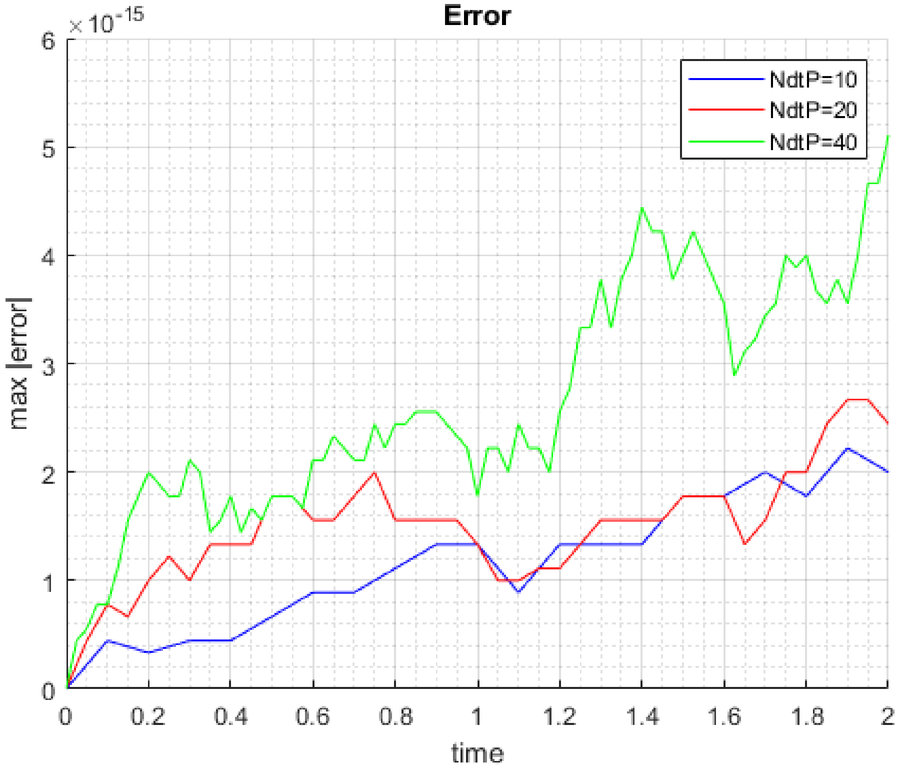

Fig. 10 Comparison of errors for D-GCL for different number of time steps per period (NdtP)

Similarly to the previous exercise, the error for each time step is sufficiently close to 0 to assume a uniform solution. The error increases with number of steps taken, which is due to round-off error and not due to a wrong implementation of the D-GCL.

References

[1] van Zuijlen, A., "Computation of Swept surfaces and volumes," , 2021.